Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	26	(build$3 or creat$3) same "data structure" same identif$5 same element$ same (set near2 element$1) and @ad<"20010117"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 16:41
L3	9	1 and memor$3 near5 structure$	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 16:41
L4	0	1 and root adj5 memor$3 and memor$3 near5 structure$	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 16:23
L5	0	3 and root adj5 memor$3 and memor$3 near5 structure$	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 16:23
L6	3	3 and root	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 16:25
L7	0	6 and quer$3 near3 element$1	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 16:25
L8	1	6 and (quer$3 or search$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 16:44
L9	118	receiv$3 same information$1 same identif$5 and query$3 near2 element$ and @ad<"20010117"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 16:43
L10	36	9 and set near3 element$1	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 16:44
L11	36	10 and (quer$3 or search$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 16:46
L12	36	10 and (quer$3 or search$3) near4 element$1	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 16:46
L13	16	12 and comparison$1	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 17:23

L14	2867	707/102.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 17:23
L15	6	14 and "data structure" and "query element"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 17:25
L16	5	15 and comparison$1 and set	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/04/15 17:25

Terms used **build** or **creat** **data** **structure** and **sets** **of** **elements** Found **99,833** of **153,034**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 Out-of-core build of a topological data structure from polygon soup

Sara McMains, Joseph M. Hellerstein, Carlo H. Séquin

May 2001 **Proceedings of the sixth ACM symposium on Solid modeling and applications**

Full text available: pdf(1.22 MB) Additional Information: full citation, abstract, references, citings, index terms

Many solid modeling applications require information not only about the geometry of an object but also about its topology. Most interchange formats do not provide this information, which the application must then derive as it builds its own topological data structure from unordered, "polygon soup" input. For very large data sets, the topological data structure itself can be bigger than core memory, so that a naive algorithm for building it that doesn't take virtual memory access p ...

2 A Survey of Data Structures for Computer Graphics Systems

Robin Williams

January 1971 **ACM Computing Surveys (CSUR)**, Volume 3 Issue 1

Full text available: pdf(1.67 MB) Additional Information: full citation, abstract, references, citings, index terms

This is a survey of a data structures and their use in computer graphics systems. First, the reasons for using data structures are given. Then the sequential, random, and list organizations are discussed, and it is shown how they may be used to build complex data structures. Representative samples of languages specifically designed for creating and manipulating data structures are described next. Finally some typical computer graphics systems and their data structures are described. It is a ...

3 Fast detection of communication patterns in distributed executions

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Full text available: pdf(4.21 MB) Additional Information: full citation, abstract, references, index terms

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

4 Streams, structures, spaces, scenarios, societies (5s): A formal model for digital libraries

Marcos André Gonçalves, Edward A. Fox, Layne T. Watson, Neill A. Kipp
April 2004 **ACM Transactions on Information Systems (TOIS)**, Volume 22 Issue 2

Full text available: pdf(316.85 KB) Additional Information: full citation, abstract, references, citings, index terms, review

Digital libraries (DLs) are complex information systems and therefore demand formal foundations lest development efforts diverge and interoperability suffers. In this article, we propose the fundamental abstractions of Streams, Structures, Spaces, Scenarios, and Societies (5S), which allow us to define digital libraries rigorously and usefully. Streams are sequences of arbitrary items used to describe both static and dynamic (e.g., video) content. Structures can be viewed as labeled directed gra ...

Keywords: applications., definitions, foundations, taxonomy

5 The Quadtree and Related Hierarchical Data Structures

Hanan Samet
June 1984 **ACM Computing Surveys (CSUR)**, Volume 16 Issue 2

Full text available: pdf(4.87 MB) Additional Information: full citation, references, citings, index terms

6 Special issue: AI in engineering

D. Sriram, R. Joobbani
January 1985 **ACM SIGART Bulletin**, Issue 91

Full text available: pdf(8.79 MB) Additional Information: full citation, abstract

The papers in this special issue were compiled from responses to the announcement in the July 1984 issue of the SIGART newsletter and notices posted over the ARPAnet. The interest being shown in this area is reflected in the sixty papers received from over six countries. About half the papers were received over the computer network.

7 A facility for defining and manipulating generalized data structures

Billy G. Claybrook
December 1977 **ACM Transactions on Database Systems (TODS)**, Volume 2 Issue 4

Full text available: pdf(2.87 MB) Additional Information: full citation, abstract, references, citings, index terms

A data structure definition facility (DSDF) is described that provides definitions for several primitive data types, homogeneous and heterogeneous arrays, cells, stacks, queues, trees, and general lists. Each nonprimitive data structure consists of two separate entities—a head and a body. The head contains the entry point(s) to the body of the structure; by treating the head like a cell, the DSDF operations are capable of creating and manipulating very general data structures. A templ ...

Keywords: data definition languages, data structure definition facility, data structures, database management

8 Software reuse

Charles W. Krueger
June 1992 **ACM Computing Surveys (CSUR)**, Volume 24 Issue 2

Full text available: pdf(4.96 MB) Additional Information: full citation, abstract, references, citings, index terms

Software reuse is the process of creating software systems from existing software rather than building software systems from scratch. This simple yet powerful vision was introduced in 1968. Software reuse has, however, failed to become a standard software engineering practice. In an attempt to understand why, researchers have renewed their interest in software reuse and in the obstacles to implementing it. This paper surveys the different approaches to software reuse found in the ...

Keywords: abstraction, cognitive distance, software reuse

9 External memory algorithms and data structures: dealing with massive data

Jeffrey Scott Vitter

June 2001 **ACM Computing Surveys (CSUR)**, Volume 33 Issue 2

Full text available: pdf(628.46 KB) Additional Information: full citation, abstract, references, citings, index terms

Data sets in large applications are often too massive to fit completely inside the computers internal memory. The resulting input/output communication (or I/O) between fast internal memory and slower external memory (such as disks) can be a major performance bottleneck. In this article we survey the state of the art in the design and analysis of external memory (or EM) algorithms and data structures, where the goal is to exploit locality in order to reduce the I/O costs. We consider a varie ...

Keywords: B-tree, I/O, batched, block, disk, dynamic, extendible hashing, external memory, hierarchical memory, multidimensional access methods, multilevel memory, online, out-of-core, secondary storage, sorting

10 An XML query engine for network-bound data

Zachary G. Ives, A. Y. Halevy, D. S. Weld

December 2002 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 11 Issue 4

Full text available: pdf(351.86 KB) Additional Information: full citation, abstract, citings, index terms

XML has become the lingua franca for data exchange and integration across administrative and enterprise boundaries. Nearly all data providers are adding XML import or export capabilities, and standard XML Schemas and DTDs are being promoted for all types of data sharing. The ubiquity of XML has removed one of the major obstacles to integrating data from widely disparate sources - namely, the heterogeneity of data formats. However, general-purpose integration of data across the wide are a also re ...

Keywords: Data integration, Data streams, Query processing, Web and databases, XML

11 Computing curricula 2001

September 2001 **Journal on Educational Resources in Computing (JERIC)**

Full text available: pdf(613.63 KB) html(2.78 KB) Additional Information: full citation, references, citings, index terms

12 Reusable Ada libraries supporting infinite data structures

Arthur G. Duncan

November 1998 **ACM SIGAda Ada Letters , Proceedings of the 1998 annual ACM SIGAda**

international conference on Ada, Volume XVIII Issue 6
Full text available: pdf(1.49 MB) Additional Information: full citation, references, index terms

Keywords: Ada, functional programming, infinite data structures, lazy evaluation

13 IS '97: model curriculum and guidelines for undergraduate degree programs in information systems
Gordon B. Davis, John T. Gorgone, J. Daniel Couger, David L. Feinstein, Herbert E. Longenecker
December 1996 **ACM SIGMIS Database , Guidelines for undergraduate degree programs on Model curriculum and guidelines for undergraduate degree programs in information systems**, Volume 28 Issue 1
Full text available: pdf(7.24 MB) Additional Information: full citation, citings

14 Session 1A: The Bloomier filter: an efficient data structure for static support lookup tables
Bernard Chazelle, Joe Kilian, Ronitt Rubinfeld, Ayellet Tal
January 2004 **Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms**
Full text available: pdf(281.39 KB) Additional Information: full citation, abstract, references

We introduce the *Bloomier filter*, a data structure for compactly encoding a function with static support in order to support approximate evaluation queries. Our construction generalizes the classical Bloom filter, an ingenious hashing scheme heavily used in networks and databases, whose main attribute---space efficiency---is achieved at the expense of a tiny false-positive rate. Whereas Bloom filters can handle only set membership queries, our Bloomier filters can deal with arbitrary func ...

15 Pointer analysis for structured parallel programs
Radu Rugina, Martin C. Rinard
January 2003 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 25 Issue 1
Full text available: pdf(383.29 KB) Additional Information: full citation, abstract, references, index terms

This paper presents a novel interprocedural, flow-sensitive, and context-sensitive pointer analysis algorithm for multithreaded programs that may concurrently update shared pointers. The algorithm is designed to handle programs with structured parallel constructs, including fork-join constructs, parallel loops, and conditionally spawned threads. For each pointer and each program point, the algorithm computes a conservative approximation of the memory locations to which that pointer may point. Th ...

Keywords: Pointer analysis

16 Research sessions: schema discovery: Information-theoretic tools for mining database structure from large data sets
Periklis Andritsos, Renée J. Miller, Panayiotis Tsaparas
June 2004 **Proceedings of the 2004 ACM SIGMOD international conference on Management of data**
Full text available: pdf(217.06 KB) Additional Information: full citation, abstract, references

Data design has been characterized as a process of arriving at a design that maximizes the

information content of each piece of data (or equivalently, one that minimizes redundancy). Information content (or redundancy) is measured with respect to a prescribed model for the data, a model that is often expressed as a set of constraints. In this work, we consider the problem of doing data redesign in an environment where the prescribed model is unknown or incomplete. Specifically, we consider the p ...

17 On understanding types, data abstraction, and polymorphism
Luca Cardelli, Peter Wegner
December 1985 **ACM Computing Surveys (CSUR)**, Volume 17 Issue 4

Full text available: pdf(4.20 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Our objective is to understand the notion of *type* in programming languages, present a model of typed, polymorphic programming languages that reflects recent research in type theory, and examine the relevance of recent research to the design of practical programming languages. Object-oriented languages provide both a framework and a motivation for exploring the interaction among the concepts of type, data abstraction, and polymorphism, since they extend the notion of type t ...

18 Data structures and algorithms for disjoint set union problems
Zvi Galil, Giuseppe F. Italiano
September 1991 **ACM Computing Surveys (CSUR)**, Volume 23 Issue 3

Full text available: pdf(2.31 MB) Additional Information: full citation, references, citings, index terms, review

Keywords: equivalence algorithm, partition, set union, time complexity

19 Special issue: Game-playing programs: theory and practice
M. A. Bramer
April 1972 **ACM SIGART Bulletin**, Issue 80

Full text available: pdf(9.23 MB) Additional Information: full citation, abstract

This collection of articles has been brought together to provide SIGART members with an overview of Artificial Intelligence approaches to constructing game-playing programs. Papers on both theory and practice are included.

20 Formal interpreters for diagram notations
Luciano Baresi, Mauro Pezzè
January 2005 **ACM Transactions on Software Engineering and Methodology (TOSEM)**, Volume 14 Issue 1

Full text available: pdf(834.85 KB) Additional Information: full citation, abstract, references, index terms

The article proposes an approach for defining extensible and flexible formal interpreters for diagram notations with significant dynamic semantics. More precisely, it addresses semi-formal diagram notations that have precisely-defined syntax, but informally defined (dynamic) semantics. These notations are often flexible to fit the different needs and expectations of users. Flexibility comes from the incompleteness or informality of the original definition and results in different interpretations ...

Keywords: Semi-formal notations, graph transformation, high-level Petri nets, semantics

Terms used
data structure and **sets of elements** and **memory structure**

Found **97,981** of **153,034**

Sort results by: relevance
Display results: expanded form
Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown

Relevance scale

1 External memory algorithms and data structures: dealing with **massive data**
Jeffrey Scott Vitter
June 2001 **ACM Computing Surveys (CSUR)**, Volume 33 Issue 2

Full text available: pdf(828.46 KB) Additional Information: full citation, abstract, references, citings, index terms

Data sets in large applications are often too massive to fit completely inside the computers internal memory. The resulting input/output communication (or I/O) between fast internal memory and slower external memory (such as disks) can be a major performance bottleneck. In this article we survey the state of the art in the design and analysis of external memory (or EM) algorithms and data structures, where the goal is to exploit locality in order to reduce the I/O costs. We consider a varie ...

Keywords: B-tree, I/O, batched, block, disk, dynamic, extendible hashing, external memory, hierarchical memory, multidimensional access methods, multilevel memory, online, out-of-core, secondary storage, sorting

2 Supporting shared data structures on distributed memory architectures
C. Koelbel, P. Mehrotra, J. Van Rosendale
February 1990 **ACM SIGPLAN Notices , Proceedings of the second ACM SIGPLAN symposium on Principles & practice of parallel programming**, Volume 25 Issue 3

Full text available: pdf(1.14 MB) Additional Information: full citation, abstract, references, citings, index terms

Programming nonshared memory systems is more difficult than programming shared memory systems, since there is no support for shared data structures. Current programming languages for distributed memory architectures force the user to decompose all data structures into separate pieces, with each piece "owned" by one of the processors in the machine, and with all communication explicitly specified by low-level message-passing primitives. This paper presents a new programming envir ...

3 A Survey of Data Structures for Computer Graphics Systems
Robin Williams
January 1971 **ACM Computing Surveys (CSUR)**, Volume 3 Issue 1

Full text available: pdf(1.67 MB) Additional Information: full citation, abstract, references, citings, index

terms

This is a survey of a data structures and their use in computer graphics systems. First, the reasons for using data structures are given. Then the sequential, random, and list organizations are discussed, and it is shown how they may be used to build complex data structures. Representative samples of languages specifically designed for creating and manipulating data structures are described next. Finally some typical computer graphics systems and their data structures are described. It is a ...

4 Automatic data structure selection: an example and overview

James R. Low

May 1978 **Communications of the ACM**, Volume 21 Issue 5

Full text available: pdf(1.14 MB) Additional Information: full citation, abstract, references, citings, index terms

The use of several levels of abstraction has proved to be very helpful in constructing and maintaining programs. When programs are designed with abstract data types such as sets and lists, programmer time can be saved by automating the process of filling in low-level implementation details. In the past, programming systems have provided only a single general purpose implementation for an abstract type. Thus the programs produced using abstract types were often inefficient in space or time. ...

Keywords: abstract data types, automatic programming, data structures, lists, optimizing compilers, sets

5 Anti-presistence: history independent data structures

Moni Naor, Vanessa Teague

July 2001 **Proceedings of the thirty-third annual ACM symposium on Theory of computing**

Full text available: pdf(308.07 KB) Additional Information: full citation, abstract, references, citings, index terms

Many data structures give away much more information than they were intended to. Whenever privacy is important, we need to be concerned that it might be possible to infer information from the memory representation of a data structure that is not available through its "legitimate" interface. Word processors that quietly maintain old versions of a document are merely the most egregious example of a general problem.We deal with data structures whose current memory repres ...

Keywords: algorithms, anti-persistence, data structures, hash table, history independence, privacy, security

6 Data structures and algorithms for disjoint set union problems

Zvi Galil, Giuseppe F. Italiano

September 1991 **ACM Computing Surveys (CSUR)**, Volume 23 Issue 3

Full text available: pdf(2.31 MB) Additional Information: full citation, references, citings, index terms, review

Keywords: equivalence algorithm, partition, set union, time complexity

7 Automatic detection and repair of errors in data structures

Brian Demsky, Martin Rinard

October 2003 **ACM SIGPLAN Notices , Proceedings of the 18th annual ACM SIGPLAN**

conference on Object-oriented programing, systems, languages, and applications, Volume 38 Issue 11

Full text available: pdf(340.56 KB) Additional Information: full citation, abstract, references, citings, index terms

We present a system that accepts a specification of key data structure consistency constraints, then dynamically detects and repairs violations of these constraints, enabling the program to continue to execute productively even in the face of otherwise crippling errors. Our experience using our system indicates that the specifications are relatively easy to develop once one understands the data structures. Furthermore, for our set of benchmark applications, our system can effectively repair inco ...

Keywords: data structure invariants, data structure repair

8 Out-of-core build of a topological data structure from polygon soup

Sara McMains, Joseph M. Hellerstein, Carlo H. Séquin

May 2001 **Proceedings of the sixth ACM symposium on Solid modeling and applications**

Full text available: pdf(1.22 MB) Additional Information: full citation, abstract, references, citings, index terms

Many solid modeling applications require information not only about the geometry of an object but also about its topology. Most interchange formats do not provide this information, which the application must then derive as it builds its own topological data structure from unordered, "polygon soup" input. For very large data sets, the topological data structure itself can be bigger than core memory, so that a naive algorithm for building it that doesn't take virtual memory access p ...

9 Supporting dynamic data structures on distributed-memory machines

Anne Rogers, Martin C. Carlisle, John H. Reppy, Laurie J. Hendren

March 1995 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 17 Issue 2

Full text available: pdf(2.05 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Compiling for distributed-memory machines has been a very active research area in recent years. Much of this work has concentrated on programs that use arrays as their primary data structures. To date, little work has been done to address the problem of supporting programs that use pointer-based dynamic data structures. The techniques developed for supporting SPMD execution of array-based programs rely on the fact that arrays are statically defined and directly addressable. Recursive data s ...

Keywords: dynamic data structures

10 Data structure architectures - a major operational principle

W. K. Giloi, H. K. Berg

April 1978 **Proceedings of the 5th annual symposium on Computer architecture**

Full text available: pdf(686.80 KB) Additional Information: full citation, abstract, references, citings, index terms

Computer architectures may be characterized by their operational principle and their physical structure. The paper defines these two characteristics for the novel concept of data structure architectures (DSAs). The representation and processing of arbitrary data structures in such a DSA is demonstrated by examples. It is shown how the functional requirements of a DSA can be satisfied by the specific information structure and the physical structure of the STARLET architecture introduced in p ...

11 Array regrouping and structure splitting using whole-program reference affinity

Yutao Zhong, Maksim Orlovich, Xipeng Shen, Chen Ding

June 2004 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2004 conference on Programming language design and implementation**, Volume 39 Issue 6

Full text available: pdf(202.16 KB) Additional Information: full citation, abstract, references, citings, index terms

While the memory of most machines is organized as a hierarchy, program data are laid out in a uniform address space. This paper defines a model of *reference affinity*, which measures how close a group of data are accessed together in a reference trace. It proves that the model gives a hierarchical partition of program data. At the top is the set of all data with the weakest affinity. At the bottom is each data element with the strongest affinity. Based on the theoretical model, the paper p ...

Keywords: array regrouping, program locality, program transformation, reference affinity, reuse signature, structure splitting, volume distance

12 The cell probe complexity of dynamic data structures

M. Fredman, M. Saks

February 1989 **Proceedings of the twenty-first annual ACM symposium on Theory of computing**

Full text available: pdf(1.23 MB) Additional Information: full citation, abstract, references, citings, index terms

Dynamic data structure problems involve the representation of data in memory in such a way as to permit certain types of modifications of the data (updates) and certain types of questions about the data (queries). This paradigm encompasses many fundamental problems in computer science. The purpose of this paper is to prove new lower and upper bounds on the time per operation to implement solutions to some familiar dynamic data structure problems including list representation, sub ...

13 Software transactional memory for dynamic-sized data structures

Maurice Herlihy, Victor Luchangco, Mark Moir, William N. Scherer

July 2003 **Proceedings of the twenty-second annual symposium on Principles of distributed computing**

Full text available: pdf(1.11 MB) Additional Information: full citation, abstract, references, citings, index terms

We propose a new form of software transactional memory (STM) designed to support dynamic-sized data structures, and we describe a novel non-blocking implementation. The non-blocking property we consider is *obstruction-freedom*. Obstruction-freedom is weaker than lock-freedom; as a result, it admits substantially simpler and more efficient implementations. A novel feature of our obstruction-free STM implementation is its use of modular contention managers to ensure progress in practice. We ...

14 A logical approach to data structures

Russell Turpin

December 1993 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 1st ACM SIGSOFT symposium on Foundations of software engineering**, Volume 18 Issue 5

Full text available: pdf(1.07 MB) Additional Information: full citation, abstract, references, index terms

The Galois project at the University of Texas is building a programming environment that supports the formal development and verification of data structure programs. This programming environment supports features such as pointer manipulation and destructive update that often make formal treatment difficult.

15 Techniques for the automatic selection of data structures
James Low, Paul Rovner
January 1976 **Proceedings of the 3rd ACM SIGACT-SIGPLAN symposium on Principles on programming languages**

Full text available: pdf(780.37 KB) Additional Information: full citation, abstract, references, citings, index terms

We are all aware of the development of increasingly sophisticated, elaborate, and expensive computer programs, particularly in the fields of artificial intelligence, data base management, and intelligent systems. The need for techniques to deal with such complexity has renewed interest in programming language research. Recent work on structured programming, intelligent compilers, automatic program generation and verification, and high-level optimization has resulted. A pattern of ...

16 Pointer analysis for structured parallel programs
Radu Rugina, Martin C. Rinard
January 2003 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 25 Issue 1

Full text available: pdf(383.29 KB) Additional Information: full citation, abstract, references, index terms

This paper presents a novel interprocedural, flow-sensitive, and context-sensitive pointer analysis algorithm for multithreaded programs that may concurrently update shared pointers. The algorithm is designed to handle programs with structured parallel constructs, including fork-join constructs, parallel loops, and conditionally spawned threads. For each pointer and each program point, the algorithm computes a conservative approximation of the memory locations to which that pointer may point. Th ...

Keywords: Pointer analysis

17 A general data dependence test for dynamic, pointer-based data structures
Joseph Hummel, Laurie J. Hendren, Alexandru Nicolau
June 1994 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1994 conference on Programming language design and implementation**, Volume 29 Issue 6

Full text available: pdf(1.27 MB) Additional Information: full citation, abstract, references, citings, index terms

Optimizing compilers require accurate dependence testing to enable numerous, performance-enhancing transformations. However, data dependence testing is a difficult problem, particularly in the presence of pointers. Though existing approaches work well for pointers to named memory locations (i.e. other variables), they are overly conservative in the case of pointers to unnamed memory locations. The latter occurs in the context of dynamic, pointer-based data structures, used in a variety of a ...

18 The Quadtree and Related Hierarchical Data Structures
Hanan Samet
June 1984 **ACM Computing Surveys (CSUR)**, Volume 16 Issue 2

Full text available: pdf(4.87 MB) Additional Information: full citation, references, citings, index terms

19 Dependence based prefetching for linked data structures
Amir Roth, Andreas Moshovos, Gurindar S. Sohi
October 1998 **Proceedings of the eighth international conference on Architectural support for programming languages and operating systems**, Volume 32 , 33 Issue 5 , 11

Full text available: pdf(1.81 MB) Additional Information: full citation, abstract, references, citings, index terms

We introduce a dynamic scheme that captures the accesspat-terns of linked data structures and can be used to predict future accesses with high accuracy. Our technique exploits the dependence relationships that exist between loads that produce addresses and loads that consume these addresses. By identzj+ing producer-consumer pairs, we construct a compact internal representation for the associated structure and its traversal. To achieve a prefetching eflect, a small prefetch engine speculatively t ...

20 Burst tries: a fast, efficient data structure for string keys

April 2002 **ACM Transactions on Information Systems (TOIS)**, Volume 20 Issue 2

Full text available: pdf(324.84 KB) Additional Information: full citation, abstract, references, citings, index terms, review

Many applications depend on efficient management of large sets of distinct strings in memory. For example, during index construction for text databases a record is held for each distinct word in the text, containing the word itself and information such as counters. We propose a new data structure, the burst trie, that has significant advantages over existing options for such applications: it uses about the same memory as a binary search tree; it is as fast as a trie; and, while not as fast as a ...

Keywords: Binary trees, splay trees, string data structures, text databases, tries, vocabulary accumulation

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Terms used **data structure** and **query elements** Found **90,194** of **153,034**

Sort results by: relevance
Display results: expanded form
Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown Relevance scale

1 Session 4A: Retroactive data structures
Erik D. Demaine, John Iacono, Stefan Langerman
January 2004 **Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms**
Full text available: pdf(163.43 KB) Additional Information: full citation, abstract, references

We introduce a new data structuring paradigm in which operations can be performed on a data structure not only in the present but also in the past. In this new paradigm, called *retroactive data structures*, the historical sequence of operations performed on the data structure is not fixed. The data structure allows arbitrary insertion and deletion of operations at arbitrary times, subject only to consistency requirements. We initiate the study of retroactive data structures by formally def ...

2 Paper session 4: XML query processing: Twig query processing over graph-structured XML data
Zografoula Vagena, Mirella M. Moro, Vassilis J. Tsotras
June 2004 **Proceedings of the 7th International Workshop on the Web and Databases: colocated with ACM SIGMOD/PODS 2004**
Full text available: pdf(522.10 KB) Additional Information: full citation, abstract, references, index terms

XML and semi-structured data is usually modeled using graph structures. *Structural summaries*, which have been proposed to speedup XML query processing have graph forms as well. The existent approaches for evaluating queries over tree structured data (i.e. data whose underlying structure is a tree) are not directly applicable when the data is modeled as a random graph. Moreover, they cannot be applied when *structural summaries* are employed and, to the best of our knowledge, no analo ...

3 Internet applications: Load balancing and locality in range-queriable data structures
James Aspnes, Jonathan Kirsch, Arvind Krishnamurthy
July 2004 **Proceedings of the twenty-third annual ACM symposium on Principles of distributed computing**
Full text available: pdf(236.00 KB) Additional Information: full citation, abstract, references, index terms

We describe a load-balancing mechanism for assigning elements to servers in a distributed data structure that supports range queries. The mechanism ensures both load-balancing with respect to an arbitrary load measure specified by the user and geographical locality, assigning elements with similar keys to the same server. Though our mechanism is specifically designed to improve the performance of skip graphs, it can be adapted to provide deterministic, locality-preserving load-balancing to any d ...

Keywords: overlay networks, peer-to-peer systems

4 Data management systems: File structure determination

Anthony J. Winkler, Alfred G. Dale

April 1971 **Proceedings of the 1971 international ACM SIGIR conference on Information storage and retrieval**

Full text available: pdf(660.48 KB) Additional Information: full citation, abstract, references

An approach to determining an appropriate file structure for a given application is presented, by outlining a methodology for comparing some important aspects of data management system performance. The aspect chosen for analysis is the processing time required to evaluate Boolean functions defined on data values contained within a file structure and select elements from the structure satisfying the expression.Two file structures are studied. The structures are each combinations of hierarchical a ...

Keywords: Boolean function evaluation, data management system, data structure, file structure, file structure determination, hierarchical file, hierarchy, inverted file, logical entry, modeling, processing algorithm, retrieval, simulation, storage structure, update

5 Computing graphical queries over XML data

Sara Comai, Ernesto Damiani, Piero Fraternali

October 2001 **ACM Transactions on Information Systems (TOIS)**, Volume 19 Issue 4

Full text available: pdf(707.80 KB) Additional Information: full citation, abstract, references, citings, index terms

The rapid evolution of XML from a mere data exchange format to a universal syntax for encoding domain-specific information raises the need for new query languages specifically conceived to address the characteristics of XML. Such languages should be able not only to extract information from XML documents, but also to apply powerful transformation and restructuring operators, based on a well-defined semantics. Moreover, XML queries should be natural to write and understand, as nontechnical person ...

Keywords: Document restructuring, graphical query languages, semantics

6 Querying structured documents with hypertext links using OODBMS

V. Christophides, A. Rizk

September 1994 **Proceedings of the 1994 ACM European conference on Hypermedia technology**

Full text available: pdf(1.32 MB) Additional Information: full citation, abstract, references, citings, index terms

Hierarchical logical structure and hypertext links are complementary and can be combined to build more powerful document management systems. Previous work exploits this complementarity for building better document processors, browsers and editing tools, but not for building sophisticated querying mechanisms. Querying in hypertext has been a requirement since [19] and has already been elaborated in many hypertext systems, but has not yet been used for hypertext systems superimposed on an und ...

Keywords: hypertexts, information retrieval, object oriented databases, path expressions, query languages, structured documents

7 An XML query engine for network-bound data

Zachary G. Ives, A. Y. Halevy, D. S. Weld
December 2002 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 11 Issue 4
Full text available: pdf(351.86 KB) Additional Information: full citation, abstract, citings, index terms

XML has become the lingua franca for data exchange and integration across administrative and enterprise boundaries. Nearly all data providers are adding XML import or export capabilities, and standard XML Schemas and DTDs are being promoted for all types of data sharing. The ubiquity of XML has removed one of the major obstacles to integrating data from widely disparate sources - namely, the heterogeneity of data formats. However, general-purpose integration of data across the wide are a also re ...

Keywords: Data integration, Data streams, Query processing, Web and databases, XML

8 XML data modeling and storage: *XVerter*: querying XML data with OR-DBMS
Humberto Vieira, Gabriela Ruberg, Marta Mattoso
November 2003 **Proceedings of the 5th ACM international workshop on Web information and data management**
Full text available: pdf(277.96 KB) Additional Information: full citation, abstract, references, index terms

Storage techniques and queries over XML databases are being widely studied. Most works store XML documents in traditional DBMSs in order to take advantage of a well established technology and also to store both structured data and XML data within a single system. This work proposes a translation mechanism to execute queries expressed on XQuery on top of XML documents that are stored in an object DBMS using the DOM implementation in disk. Rules for automatic translation from XQuery to SQL3 are pr ...

Keywords: DOM, SQL3, XML, XQuery, XSLT, object DBMS

9 XML query processing: Efficient evaluation of multiple queries on streaming XML data
Mong Li Lee, Boon Chin Chua, Wynne Hsu, Kian-Lee Tan
November 2002 **Proceedings of the eleventh international conference on Information and knowledge management**
Full text available: pdf(427.96 KB) Additional Information: full citation, abstract, references, index terms

Traditionally, XML documents are processed at where they are stored. This allows the query processor to exploit pre-computed data structures (e.g., index) to retrieve the desired data efficiently. However, this mode of processing is not suitable for many applications where the documents are frequently updated. In such situations, efficient evaluation of multiple queries over streaming XML documents becomes important. This paper introduces a new operator, mqX-scan, which efficiently evaluates mul ...

Keywords: finite state machines, regular path expressions, streaming XML

10 XML and text: Querying structured text in an XML database
Shurug Al-Khalifa, Cong Yu, H. V. Jagadish
June 2003 **Proceedings of the 2003 ACM SIGMOD international conference on Management of data**
Full text available: pdf(242.55 KB) Additional Information: full citation, abstract, references, citings, index terms

XML databases often contain documents comprising structured text. Therefore, it is important to integrate "information retrieval style" query evaluation, which is well-suited for natural language text, with standard "database style" query evaluation, which handles structured queries efficiently. Relevance scoring is central to information retrieval. In the

case of XML, this operation becomes more complex because the data required for scoring could reside not directly in an element itself but als ...

11 Proximal nodes: a model to query document databases by content and structure
Gonzalo Navarro, Ricardo Baeza-Yates
October 1997 **ACM Transactions on Information Systems (TOIS)**, Volume 15 Issue 4

Full text available: pdf(550.43 KB) Additional Information: full citation, abstract, references, citings, index terms, review

A model to query document databases by both their content and structure is presented. The goal is to obtain a query language that is expressive in practice while being efficiently implementable, features not present at the same time in previous work. The key ideas of the model are a set-oriented query language based on operations on nearby structure elements of one or more hierarchies, together with content and structural indexing and bottom-up evaluation. The model is evaluated in regard t ...

Keywords: expressivity and efficiency of query languages, hierarchical documents, structured text, text algebras

12 Graph-based GUIs for querying XML data: the XML-GL experience
S. Comai
March 2001 **Proceedings of the 2001 ACM symposium on Applied computing**

Full text available: pdf(202.95 KB) Additional Information: full citation, references, index terms

13 XML indexing and compression: ViST: a dynamic index method for querying XML data by tree structures
Haixun Wang, Sanghyun Park, Wei Fan, Philip S. Yu
June 2003 **Proceedings of the 2003 ACM SIGMOD international conference on Management of data**

Full text available: pdf(244.47 KB) Additional Information: full citation, abstract, references, citings, index terms

With the growing importance of XML in data exchange, much research has been done in providing flexible query facilities to extract data from structured XML documents. In this paper, we propose ViST, a novel index structure for searching XML documents. By representing both XML documents and XML queries in structure-encoded sequences, we show that querying XML data is equivalent to finding subsequence matches. Unlike index methods that disassemble a query into multiple sub-queries, and then *joi* ...

14 Ad Hoc Query: a reusable database access capability
J. Wolfe
July 1994 **Proceedings of the eleventh annual Washington Ada symposium & summer ACM SIGAda meeting on Ada**

Full text available: pdf(1.06 MB) Additional Information: full citation, references, index terms

15 Streaming XML: XPath queries on streaming data
Feng Peng, Sudarshan S. Chawathe
June 2003 **Proceedings of the 2003 ACM SIGMOD international conference on Management of data**

Full text available: pdf(433.73 KB) Additional Information: full citation, abstract, references, citings, index terms

We present the design and implementation of the XSQ system for querying streaming XML data using XPath 1.0. Using a clean design based on a hierarchical arrangement of pushdown transducers augmented with buffers, XSQ supports features such as multiple predicates, closures, and aggregation. XSQ not only provides high throughput, but is also memory efficient: It buffers only data that must be buffered by any streaming XPath processor. We also present an empirical study of the performance character ...

16 An overview and classification of mediated query systems 
Ruxandra Domenig, Klaus R. Dittrich
September 1999 **ACM SIGMOD Record**, Volume 28 Issue 3

Full text available: pdf(397.64 KB) Additional Information: full citation, abstract, citings, index terms

Multimedia technology, global information infrastructures and other developments allow users to access more and more information sources of various types. However, the "technical" availability alone (by means of networks, WWW, mail systems, databases, etc.) is not sufficient for making meaningful and advanced use of all information available on-line. Therefore, the problem of effectively and efficiently accessing and querying heterogeneous and distributed data sources is an impo ...

17 Quasi-Valid range querying and its implications for nearest neighbor problems 
D. E. Willard, Y. C. Wee
January 1988 **Proceedings of the fourth annual symposium on Computational geometry**

Full text available: pdf(654.95 KB) Additional Information: full citation, abstract, references, citings, index terms

We define a new formalism called the quasi-valid range aggregation. This formalism leads to a new and quite simple method for reducing non-range query-like problems to range queries and often to orthogonal range queries, with immediate applications to the ATTRACTED NEIGHBOR and the planar ALL-PAIRS NEAREST NEIGHBORS problems (the latter being solved optimally on both parallel and sequential machines). A point of special interest is that our new formalism permits operators "+" th ...

18 XML and semistructured data querying: Measuring similarity between collection of values 
Carina F. Dorneles, Carlos A. Heuser, Andrei E. N. Lima, Altigran Soares da Silva, Edleno Silva de Moura
November 2004 **Proceedings of the 6th annual ACM international workshop on Web information and data management**

Full text available: pdf(264.68 KB) Additional Information: full citation, abstract, references, index terms

In this paper, we propose a set of similarity metrics for manipulating collections of values occuring in XML documents.

Following the data model presented in TAX algebra, we treat an XML element as a labeled ordered rooted tree. Consider that XML nodes can be either atomic, i.e, they may contain single values such as short character strings, date, etc, or complex, i.e., nested structures that contain other nodes, we propose two types of similarity metrics: MAVs, for atomic nodes and MC ...

Keywords: XML, imprecise queries, similarity functions, vague queries

19 Research sessions: similarity and matching: Statistical synopses for graph-structured XML databases 
Neoklis Polyzotis, Minos Garofalakis
June 2002 **Proceedings of the 2002 ACM SIGMOD international conference on**

Management of data

Full text available: pdf(1.44 MB) Additional Information: full citation, abstract, references, citings, index terms

Effective support for XML query languages is becoming increasingly important with the emergence of new applications that access large volumes of XML data. All existing proposals for querying XML (e.g., XQuery) rely on a *pattern-specification language* that allows path navigation and branching through the XML data graph in order to reach the desired data elements. Optimizing such queries depends crucially on the existence of concise synopsis structures that enable accurate compile-time sele ...

20 Approximate data structures with applications

Yossi Matias, Jeffrey Scott Vitter, Neal E. Young

January 1994 **Proceedings of the fifth annual ACM-SIAM symposium on Discrete algorithms**

Full text available: pdf(938.64 KB) Additional Information: full citation, references, citings, index terms

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Terms used
data structure and **memory structure** and **root memory structure** and **location**

Found **96,939** of **153,034**

Sort results by: relevance
Display results: expanded form
Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown Relevance scale

1 External memory algorithms and data structures: dealing with massive data
Jeffrey Scott Vitter
June 2001 **ACM Computing Surveys (CSUR)**, Volume 33 Issue 2

Full text available: pdf(828.46 KB) Additional Information: full citation, abstract, references, citings, index terms

Data sets in large applications are often too massive to fit completely inside the computers internal memory. The resulting input/output communication (or I/O) between fast internal memory and slower external memory (such as disks) can be a major performance bottleneck. In this article we survey the state of the art in the design and analysis of external memory (or EM) algorithms and data structures, where the goal is to exploit locality in order to reduce the I/O costs. We consider a varie ...

Keywords: B-tree, I/O, batched, block, disk, dynamic, extendible hashing, external memory, hierarchical memory, multidimensional access methods, multilevel memory, online, out-of-core, secondary storage, sorting

2 The string B-tree: a new data structure for string search in external memory and its applications
Paolo Ferragina, Roberto Grossi
March 1999 **Journal of the ACM (JACM)**, Volume 46 Issue 2

Full text available: pdf(363.37 KB) Additional Information: full citation, abstract, references, citings, index terms

We introduce a new text-indexing data structure, the String B-Tree, that can be seen as a link between some traditional external-memory and string-matching data structures. In a short phrase, it is a combination of B-trees and Patricia tries for internal-node indices that is made more effective by adding extra pointers to speed up search and update operations. Consequently, the String B-Tree overcomes the theoretical limitations of inverted files, B-trees, prefix B-trees, s ...

Keywords: B-tree, Patricia trie, external-memory data structure, prefix and range search, string searching and sorting, suffix array, suffix tree, text index

3 The P-range tree: a new data structure for range searching in secondary memory

Sairam Subramanian, Sridhar Ramaswamy
January 1995 **Proceedings of the sixth annual ACM-SIAM symposium on Discrete algorithms**
Full text available: pdf(1.22 MB) Additional Information: full citation, references, citings, index terms

4 Supporting dynamic data structures on distributed-memory machines
Anne Rogers, Martin C. Carlisle, John H. Reppy, Laurie J. Hendren
March 1995 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 17 Issue 2
Full text available: pdf(2.05 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Compiling for distributed-memory machines has been a very active research area in recent years. Much of this work has concentrated on programs that use arrays as their primary data structures. To date, little work has been done to address the problem of supporting programs that use pointer-based dynamic data structures. The techniques developed for supporting SPMD execution of array-based programs rely on the fact that arrays are statically defined and directly addressable. Recursive data s ...

Keywords: dynamic data structures

5 A general data dependence test for dynamic, pointer-based data structures
Joseph Hummel, Laurie J. Hendren, Alexandru Nicolau
June 1994 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1994 conference on Programming language design and implementation**, Volume 29 Issue 6
Full text available: pdf(1.27 MB) Additional Information: full citation, abstract, references, citings, index terms

Optimizing compilers require accurate dependence testing to enable numerous, performance-enhancing transformations. However, data dependence testing is a difficult problem, particularly in the presence of pointers. Though existing approaches work well for pointers to named memory locations (i.e. other variables), they are overly conservative in the case of pointers to unnamed memory locations. The latter occurs in the context of dynamic, pointer-based data structures, used in a variety of a ...

6 Asynchronous shared memory search structures
Micah Adler
June 1996 **Proceedings of the eighth annual ACM symposium on Parallel algorithms and architectures**
Full text available: pdf(1.35 MB) Additional Information: full citation, references, citings, index terms

7 Burst tries: a fast, efficient data structure for string keys
April 2002 **ACM Transactions on Information Systems (TOIS)**, Volume 20 Issue 2
Full text available: pdf(324.84 KB) Additional Information: full citation, abstract, references, citings, index terms, review

Many applications depend on efficient management of large sets of distinct strings in memory. For example, during index construction for text databases a record is held for each distinct word in the text, containing the word itself and information such as counters. We propose a new data structure, the burst trie, that has significant advantages over existing options for such applications: it uses about the same memory as a binary search tree; it is as fast as a trie; and, while not as fast as a ...

Keywords: Binary trees, splay trees, string data structures, text databases, tries, vocabulary accumulation

8 Data structures: Cache-oblivious data structures for orthogonal range searching

Pankaj K. Agarwal, Lars Arge, Andrew Danner, Bryan Holland-Minkley

June 2003 **Proceedings of the nineteenth annual symposium on Computational geometry**

Full text available: pdf(353.23 KB) Additional Information: full citation, abstract, references, citings, index terms

We develop cache-oblivious data structures for orthogonal range searching, the problem of finding all T points in a set of N points in IR^d lying in a query hyper-rectangle. Cache-oblivious data structures are designed to be efficient in arbitrary memory hierarchies.We describe a dynamic linear-size data structure that answers d-dimensional queries in $O((N/B)^{1-1/d}+T/B)$ memory transfers, where B is the block size of any two levels of ...

Keywords: cache-oblivious, orthogonal range searching

9 Main-memory index structures with fixed-size partial keys

Philip Bohannon, Peter McIlroy, Rajeev Rastogi

May 2001 **ACM SIGMOD Record , Proceedings of the 2001 ACM SIGMOD international conference on Management of data**, Volume 30 Issue 2

Full text available: pdf(185.51 KB) Additional Information: full citation, abstract, references, citings, index terms

The performance of main-memory index structures is increasingly determined by the number of CPU cache misses incurred when traversing the index. When keys are stored indirectly, as is standard in main-memory databases, the cost of key retrieval in terms of cache misses can dominate the cost of an index traversal. Yet it is inefficient in both time and space to store even moderate sized keys directly in index nodes. In this paper, we investigate the performance of tree structures suitable for ...

Keywords: B-trees, T-tree, cache coherence, key compression, main-memory indices

10 The Quadtree and Related Hierarchical Data Structures

Hanan Samet

June 1984 **ACM Computing Surveys (CSUR)**, Volume 16 Issue 2

Full text available: pdf(4.87 MB) Additional Information: full citation, references, citings, index terms

11 Effective jump-pointer prefetching for linked data structures

Amir Roth, Gurindar S. Sohi

May 1999 **ACM SIGARCH Computer Architecture News , Proceedings of the 26th annual international symposium on Computer architecture**, Volume 27 Issue 2

Full text available: pdf(113.33 KB) Additional Information: full citation, abstract, references, citings, index terms

Publisher Site

Current techniques for prefetching linked data structures (LDS) exploit the work available in one loop iteration or recursive call to overlap pointer chasing latency. Jump pointers, which provide direct access to non-adjacent nodes, can be used for prefetching when loop and recursive procedure bodies are small and do not have sufficient work to overlap a long latency. This paper describes a framework for jump-pointer prefetching (JPP) that supports four prefetching idioms: queue, full, chain, an ...

12 Automatic pool allocation for disjoint data structures

Chris Lattner, Vikram Adve

June 2002 **ACM SIGPLAN Notices , Proceedings of the workshop on Memory system performance**, Volume 38 Issue 2 supplement

Full text available: pdf(1.48 MB) Additional Information: full citation, abstract, references, citings

This paper presents an analysis technique and a novel program transformation that can enable powerful optimizations for entire linked data structures. The fully automatic transformation converts ordinary programs to use pool (aka region) allocation for heap-based data structures. The transformation relies on an efficient link-time interprocedural analysis to identify disjoint data structures in the program, to check whether these data structures are accessed in a type-safe manner, and to constru ...

13 Out-of-core build of a topological data structure from polygon soup

Sara McMains, Joseph M. Hellerstein, Carlo H. Séquin

May 2001 **Proceedings of the sixth ACM symposium on Solid modeling and applications**

Full text available: pdf(1.22 MB) Additional Information: full citation, abstract, references, citings, index terms

Many solid modeling applications require information not only about the geometry of an object but also about its topology. Most interchange formats do not provide this information, which the application must then derive as it builds its own topological data structure from unordered, "polygon soup" input. For very large data sets, the topological data structure itself can be bigger than core memory, so that a naive algorithm for building it that doesn't take virtual memory access p ...

14 A structural view of the Cedar programming environment

Daniel C. Swinehart, Polle T. Zellweger, Richard J. Beach, Robert B. Hagmann

August 1986 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 8 Issue 4

Full text available: pdf(6.32 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents an overview of the Cedar programming environment, focusing on its overall structure—that is, the major components of Cedar and the way they are organized. Cedar supports the development of programs written in a single programming language, also called Cedar. Its primary purpose is to increase the productivity of programmers whose activities include experimental programming and the development of prototype software systems for a high-performance personal computer. T ...

15 Software transactional memory for dynamic-sized data structures

Maurice Herlihy, Victor Luchangco, Mark Moir, William N. Scherer

July 2003 **Proceedings of the twenty-second annual symposium on Principles of distributed computing**

Full text available: pdf(1.11 MB) Additional Information: full citation, abstract, references, citings, index terms

We propose a new form of software transactional memory (STM) designed to support dynamic-sized data structures, and we describe a novel non-blocking implementation. The non-blocking property we consider is *obstruction-freedom*. Obstruction-freedom is weaker than lock-freedom; as a result, it admits substantially simpler and more efficient implementations. A novel feature of our obstruction-free STM implementation is its use of modular contention managers to ensure progress in practice. We ...

16 A general framework for prefetch scheduling in linked data structures and its

application to multi-chain prefetching
Seungryul Choi, Nicholas Kohout, Sumit Pamnani, Dongkeun Kim, Donald Yeung
May 2004 **ACM Transactions on Computer Systems (TOCS)**, Volume 22 Issue 2

Full text available: pdf(2.45 MB) Additional Information: full citation, abstract, references, index terms

Pointer-chasing applications tend to traverse composite data structures consisting of multiple independent pointer chains. While the traversal of any single pointer chain leads to the serialization of memory operations, the traversal of independent pointer chains provides a source of memory parallelism. This article investigates exploiting such *interchain memory parallelism* for the purpose of memory latency tolerance, using a technique called *multi--chain prefetching*. Previous work ...

Keywords: Data prefetching, memory parallelism, pointer-chasing code

17 Computation techniques for FPGAs: Techniques for synthesizing binaries to an advanced register/memory structure
Greg Stitt, Zhi Guo, Walid Najjar, Frank Vahid
February 2005 **Proceedings of the 2005 ACM/SIGDA 13th international symposium on Field-programmable gate arrays**

Full text available: pdf(211.74 KB) Additional Information: full citation, abstract, references, index terms

Recent works demonstrate several benefits of synthesizing software binaries onto FPGA hardware, including incorporating hardware design into established software tool flows with minimal impact, porting existing binaries to FPGAs, and even dynamically synthesizing software kernels to faster FPGA coprocessors. Those works showed that standard binary decompilation methods can recover enough high-level control information to result in reasonably-efficient hardware. However, recent synthesis methods ...

Keywords: FPGA, binaries, decompilation, embedded systems, smart buffers, synthesis

18 Pointer analysis for structured parallel programs
Radu Rugina, Martin C. Rinard
January 2003 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 25 Issue 1

Full text available: pdf(383.29 KB) Additional Information: full citation, abstract, references, index terms

This paper presents a novel interprocedural, flow-sensitive, and context-sensitive pointer analysis algorithm for multithreaded programs that may concurrently update shared pointers. The algorithm is designed to handle programs with structured parallel constructs, including fork-join constructs, parallel loops, and conditionally spawned threads. For each pointer and each program point, the algorithm computes a conservative approximation of the memory locations to which that pointer may point. Th ...

Keywords: Pointer analysis

19 Compiler-based prefetching for recursive data structures
Chi-Keung Luk, Todd C. Mowry
October 1996 **Proceedings of the seventh international conference on Architectural support for programming languages and operating systems**, Volume 30 , 31 Issue 5 , 9

Full text available: pdf(1.51 MB) Additional Information: full citation, abstract, references, citings, index terms

Software-controlled data prefetching offers the potential for bridging the ever-increasing

speed gap between the memory subsystem and today's high-performance processors. While prefetching has enjoyed considerable success in array-based numeric codes, its potential in pointer-based applications has remained largely unexplored. This paper investigates compiler-based prefetching for pointer-based applications---in particular, those containing recursive data structures. We identify the fundamental ...

20 The structure of Cedar

Daniel C. Swinehart, Polle T. Zellweger, Robert B. Hagmann

June 1985 **Proceedings of the ACM SIGPLAN 85 symposium on Language issues in programming environments**, Volume 20 , 18 Issue 7 , 6

Full text available: pdf(1.79 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents an overview of the Cedar programming environment, focusing primarily on its overall structure: the major components of Cedar and the way they are organized. Cedar supports the development of programs written in a single programming language, also called Cedar. We will emphasize the extent to which the Cedar language, with runtime support, has influenced the organization, comprehensibility, and stability of Cedar. Produced in the Computer Science Laboratory (CS ...

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Terms used
data structure and **query elements** and **comparisons** and **set of elements**

Found **94,815** of **153,034**

Sort results by: relevance
Display results: expanded form
Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown Relevance scale

1 Semistructured Data: Structural proximity searching for large collections of semi-structured data
Michael Barg, Raymond K. Wong
October 2001 **Proceedings of the tenth international conference on Information and knowledge management**
Full text available: pdf(1.92 MB) Additional Information: full citation, abstract, references, citings, index terms

The richness of the XML data format allows data to be structured in a way which precisely captures the semantics required by the author. It is the structure of the data, however, which forms the basis of all XML query languages. Without at least some notion of the structure, a user cannot meaningfully query the data. This problem is compounded when one considers that heterogeneous data adhering to different schema are likely to exist in the database(s) being queried. This paper proposes a soluti ...

2 Proximal nodes: a model to query document databases by content and structure
Gonzalo Navarro, Ricardo Baeza-Yates
October 1997 **ACM Transactions on Information Systems (TOIS)**, Volume 15 Issue 4
Full text available: pdf(550.43 KB) Additional Information: full citation, abstract, references, citings, index terms, review

A model to query document databases by both their content and structure is presented. The goal is to obtain a query language that is expressive in practice while being efficiently implementable, features not present at the same time in previous work. The key ideas of the model are a set-oriented query language based on operations on nearby structure elements of one or more hierarchies, together with content and structural indexing and bottom-up evaluation. The model is evaluated in regard t ...

Keywords: expressivity and efficiency of query languages, hierarchical documents, structured text, text algebras

3 An XML query engine for network-bound data
Zachary G. Ives, A. Y. Halevy, D. S. Weld
December 2002 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 11 Issue 4
Full text available: pdf(351.86 KB) Additional Information: full citation, abstract, citings, index terms

XML has become the lingua franca for data exchange and integration across administrative and enterprise boundaries. Nearly all data providers are adding XML import or export capabilities, and standard XML Schemas and DTDs are being promoted for all types of data sharing. The ubiquity of XML has removed one of the major obstacles to integrating data from widely disparate sources - namely, the heterogeneity of data formats. However, general-purpose integration of data across the wide are a also re ...

Keywords: Data integration, Data streams, Query processing, Web and databases, XML

4 Computing graphical queries over XML data
Sara Comai, Ernesto Damiani, Piero Fraternali
October 2001 **ACM Transactions on Information Systems (TOIS)**, Volume 19 Issue 4

Full text available: pdf(707.80 KB) Additional Information: full citation, abstract, references, citings, index terms

The rapid evolution of XML from a mere data exchange format to a universal syntax for encoding domain-specific information raises the need for new query languages specifically conceived to address the characteristics of XML. Such languages should be able not only to extract information from XML documents, but also to apply powerful transformation and restructuring operators, based on a well-defined semantics. Moreover, XML queries should be natural to write and understand, as nontechnical person ...

Keywords: Document restructuring, graphical query languages, semantics

5 The Quadtree and Related Hierarchical Data Structures
Hanan Samet
June 1984 **ACM Computing Surveys (CSUR)**, Volume 16 Issue 2

Full text available: pdf(4.87 MB) Additional Information: full citation, references, citings, index terms

6 XML indexing and compression: ViST: a dynamic index method for querying XML data by tree structures
Haixun Wang, Sanghyun Park, Wei Fan, Philip S. Yu
June 2003 **Proceedings of the 2003 ACM SIGMOD international conference on Management of data**

Full text available: pdf(244.47 KB) Additional Information: full citation, abstract, references, citings, index terms

With the growing importance of XML in data exchange, much research has been done in providing flexible query facilities to extract data from structured XML documents. In this paper, we propose ViST, a novel index structure for searching XML documents. By representing both XML documents and XML queries in structure-encoded sequences, we show that querying XML data is equivalent to finding subsequence matches. Unlike index methods that disassemble a query into multiple sub-queries, and then *joi* ...

7 XIRQL: An XML query language based on information retrieval concepts
Norbert Fuhr, Kai Großjohann
April 2004 **ACM Transactions on Information Systems (TOIS)**, Volume 22 Issue 2

Full text available: pdf(281.91 KB) Additional Information: full citation, abstract, references, citings, index terms

XIRQL ("circle") is an XML query language that incorporates imprecision and vagueness for both structural and content-oriented query conditions. The corresponding uncertainty is handled by a consistent probabilistic model. The core features of XIRQL are (1) document

ranking based on index term weighting, (2) specificity-oriented search for retrieving the most relevant parts of documents, (3) datatypes with vague predicates for dealing with specific types of content and (4) structural vagueness f ...

Keywords: Path algebra, XML, XQuery, probabilistic retrieval, ranked retrieval, vague predicates

8 Data management systems: File structure determination

Anthony J. Winkler, Alfred G. Dale
April 1971 **Proceedings of the 1971 international ACM SIGIR conference on Information storage and retrieval**
Full text available: pdf(660.48 KB) Additional Information: full citation, abstract, references

An approach to determining an appropriate file structure for a given application is presented, by outlining a methodology for comparing some important aspects of data management system performance. The aspect chosen for analysis is the processing time required to evaluate Boolean functions defined on data values contained within a file structure and select elements from the structure satisfying the expression.Two file structures are studied. The structures are each combinations of hierarchical a ...

Keywords: Boolean function evaluation, data management system, data structure, file structure, file structure determination, hierarchical file, hierarchy, inverted file, logical entry, modeling, processing algorithm, retrieval, simulation, storage structure, update

9 Document management: Context representation, transformation and comparison for ad hoc product data exchange

Jingzhi Guo, Chengzheng Sun
November 2003 **Proceedings of the 2003 ACM symposium on Document engineering**
Full text available: pdf(275.65 KB) Additional Information: full citation, abstract, references, index terms

Product data exchange is the precondition of business interoperation between Web-based firms. However, millions of small and medium sized enterprises (SMEs) encode their Web product data in ad hoc formats for electronic product catalogues. This prevents product data exchange between business partners for business interoperation. To solve this problem, this paper has proposed a novel concept-centric catalogue engineering approach for representing, transforming and comparing semantic contexts in a ...

Keywords: XML product map, XPM, ad hoc product data exchange, concept, context comparison, context representation, context transformation, electronic commerce, electronic product catalogue, product data integration, semantics

10 External memory algorithms and data structures: dealing with massive data

Jeffrey Scott Vitter
June 2001 **ACM Computing Surveys (CSUR)**, Volume 33 Issue 2
Full text available: pdf(828.46 KB) Additional Information: full citation, abstract, references, citings, index terms

Data sets in large applications are often too massive to fit completely inside the computers internal memory. The resulting input/output communication (or I/O) between fast internal memory and slower external memory (such as disks) can be a major performance bottleneck. In this article we survey the state of the art in the design and analysis of external memory (or EM) algorithms and data structures, where the goal is to exploit locality in order to reduce the I/O costs. We consider a varie ...

Keywords: B-tree, I/O, batched, block, disk, dynamic, extendible hashing, external memory, hierarchical memory, multidimensional access methods, multilevel memory, online, out-of-core, secondary storage, sorting

11 Query optimization in a memory-resident domain relational calculus database system

Kyu-Young Whang, Ravi Krishnamurthy

March 1990 **ACM Transactions on Database Systems (TODS)**, Volume 15 Issue 1

Full text available: pdf(2.46 MB) Additional Information: full citation, abstract, references, citings, index terms

We present techniques for optimizing queries in memory-resident database systems. Optimization techniques in memory-resident database systems differ significantly from those in conventional disk-resident database systems. In this paper we address the following aspects of query optimization in such systems and present specific solutions for them: (1) a new approach to developing a CPU-intensive cost model; (2) new optimization strategies for main-memory query processing; (3) new insight into ...

12 A Comparison of the Relational and CODASYL Approaches to Data-Base Management

Ann S. Michaels, Benjamin Mittman, C. Robert Carlson

January 1976 **ACM Computing Surveys (CSUR)**, Volume 8 Issue 1

Full text available: pdf(2.06 MB) Additional Information: full citation, references, citings, index terms

13 RE-tree: an efficient index structure for regular expressions

Chee-Yong Chan, Minos Garofalakis, Rajeev Rastogi

August 2003 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 12 Issue 2

Full text available: pdf(346.00 KB) Additional Information: full citation, abstract, index terms

Abstract.Due to their expressive power, regular expressions (REs) are quickly becoming an integral part of language specifications for several important application scenarios. Many of these applications have to manage huge databases of RE specifications and need to provide an effective matching mechanism that, given an input string, quickly identifies the REs in the database that match it. In this paper, we propose the RE-tree, a novel index structure for large databases of RE specifications. Gi ...

Keywords: Index structure, Regular expressions, Sampling-based approximations, Size measures

14 Session 4A: Retroactive data structures

Erik D. Demaine, John Iacono, Stefan Langerman

January 2004 **Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms**

Full text available: pdf(183.43 KB) Additional Information: full citation, abstract, references

We introduce a new data structuring paradigm in which operations can be performed on a data structure not only in the present but also in the past. In this new paradigm, called *retroactive data structures*, the historical sequence of operations performed on the data structure is not fixed. The data structure allows arbitrary insertion and deletion of operations at arbitrary times, subject only to consistency requirements. We initiate the study of retroactive data structures by formally def ...

15 Query Optimization in Database Systems
Matthias Jarke, Jurgen Koch
June 1984 **ACM Computing Surveys (CSUR)**, Volume 16 Issue 2
Full text available: pdf(2.84 MB) Additional Information: full citation, references, citings, index terms

16 Deferred data structuring: Query-driven preprocessing for geometric search problems
R Motwani, P Raghavan
August 1986 **Proceedings of the second annual symposium on Computational geometry**
Full text available: pdf(672.79 KB) Additional Information: full citation, abstract, references, citings, index terms

We consider the problem of answering a series of on-line queries on a static database. The conventional approach to such problems involves a preprocessing phase which constructs a data structure with good search behavior. The data structure is then used to process a series of queries without any further reordering. Our approach involves dynamic or query-driven structuring of the database, i.e. we process the database only when it is required for answering a query. We present optimal algorit ...

17 Algebraic query optimisation for database programming languages
Alexandra Poulovassilis, Carol Small
April 1996 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 5 Issue 2
Full text available: pdf(193.10 KB) Additional Information: full citation, abstract, citings, index terms

A major challenge still facing the designers and implementors of database programming languages (DBPLs) is that of query optimisation. We investigate algebraic query optimisation techniques for DBPLs in the context of a purely declarative functional language that supports sets as first-class objects. Since the language is computationally complete issues such as non-termination of expressions and construction of infinite data structures can be investigated, whilst its declarative nature allows th ...

Keywords: Algebraic manipulation, Database management, Database programming languages, Functional languages, Query optimisation

18 Data structures and algorithms for nearest neighbor search in general metric spaces
Peter N. Yianilos
January 1993 **Proceedings of the fourth annual ACM-SIAM Symposium on Discrete algorithms**
Full text available: pdf(1.34 MB) Additional Information: full citation, references, citings, index terms

Keywords: associative memory, clustering, computational geometry, metric space, nearest neighbor, pattern recognition, randomized methods

19 A language for queries on structure and contents of textual databases
Gonzalo Navarro, Ricardo Baeza-Yates
July 1995 **Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval**
Full text available: pdf(1.05 MB) Additional Information: full citation, references, citings, index terms

20 Semantics of query languages for network databases

Kazimierz Subieta

September 1985 **ACM Transactions on Database Systems (TODS)**, Volume 10 Issue 3

Full text available: pdf(3.71 MB) Additional Information: full citation, abstract, references, index terms

Semantics determines the meaning of language constructs; hence it says much more than syntax does about implementing the language. The main purpose of this paper is a formal presentation of the meaning of basic language constructs employed in many database languages (sublanguages). Therefore, stylized query languages SSL (Sample Selection Language) and J (Joins) are introduced, wherein most of the typical entries present in other query languages are collected. The semantics of SSL and J are ...

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Results for **"(data structure<in>metadata)"**

Your search matched **6032** of **1144303** documents.

A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» View Session History

» New Search

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

Modify Search

(data structure<in>metadata)

Check to search only within this results set

Display Format: Citation Citation & Abstract

Select Article Information View: **1-25** |

1. **Automatic composition of data structures to represent relations**
Cohen, D.; Campbell, N.;
Knowledge-Based Software Engineering Conference, 1992., Proceedings of the Seventh
20-23 Sept. 1992 Page(s):182 - 191
AbstractPlus | Full Text: PDF(816 KB) IEEE CNF

2. **Using parallel data structures in optimistic discrete event simulation of varying granularity on shared**
Prasad, S.K.; Sawant, S.I.; Naqib, B.;
Algorithms and Architectures for Parallel Processing, 1995. ICAPP 95. IEEE First ICA/sup 3/PP. IEEE First In Conference on
Volume 1, 19-21 April 1995 Page(s):365 - 374 vol.1
AbstractPlus | Full Text: PDF(616 KB) IEEE CNF

3. **Design of multi-invariant data structures for robust shared accesses in multiprocessor systems**
I-Ling Yen; Bastani, F.B.; Taylor, D.J.;
Software Engineering, IEEE Transactions on
Volume 27, Issue 3, March 2001 Page(s):193 - 207
AbstractPlus | References | Full Text: PDF(276 KB) IEEE JNL

4. **The whiteboard environment: an electronic sketchpad for data structure design and algorithm descrip**
Brown, D.R.; Zanden, B.V.;
Visual Languages, 1998. Proceedings. 1998 IEEE Symposium on
1-4 Sept. 1998 Page(s):288 - 295
AbstractPlus | Full Text: PDF(52 KB) IEEE CNF

5. **Automated instrumentation and monitoring of data movement in parallel programs**
Sarukkai, S.R.; Yan, J.; Schmidt, M.;
Parallel Processing Symposium, 1995. Proceedings., 9th International
25-28 April 1995 Page(s):621 - 629
AbstractPlus | Full Text: PDF(1040 KB) IEEE CNF

6. **Performance comparison of several data structures for storing VLSI geometry**
Grgek, M.; Branica, I.; Divkovic-Puksec, J.;
EUROCON 2003. Computer as a Tool. The IEEE Region 8
Volume 1, 22-24 Sept. 2003 Page(s):156 - 159 vol.1
AbstractPlus | Full Text: PDF(362 KB) IEEE CNF

7. **SPMD execution of programs with dynamic data structures on distributed memory machines**
Gupta, R.;
Computer Languages, 1992., Proceedings of the 1992 International Conference on
20-23 April 1992 Page(s):232 - 241
AbstractPlus | Full Text: PDF(868 KB) IEEE CNF

8. **Software components in a data structure precompiler**
Sirkin, M.; Batory, D.; Singhal, V.;
Software Engineering, 1993. Proceedings., 15th International Conference on
17-21 May 1993 Page(s):437 - 446
AbstractPlus | Full Text: PDF(780 KB) IEEE CNF

9. **Efficient proximity search in multivariate data**
Kao, D.T.; Bergeron, R.D.; Sparr, T.M.;
Scientific and Statistical Database Management, 1998. Proceedings. Tenth International Conference on
1-3 July 1998 Page(s):145 - 154
AbstractPlus | Full Text: PDF(404 KB) IEEE CNF

10. **Opportunistic data structures with applications**
Ferragina, P.; Manzini, G.;
Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on
12-14 Nov. 2000 Page(s):390 - 398
AbstractPlus | Full Text: PDF(844 KB) IEEE CNF

11. **Visual editing of data structures**
Robson, R.; Seminar, K.;
Software Maintenance, 1991., Proceedings. Conference on
15-17 Oct. 1991 Page(s):228 - 237
AbstractPlus | Full Text: PDF(568 KB) IEEE CNF

12. **Synthesized data structure for video data: a formal specification**
Sridharan, P.N.; Raman, S.;
Image Processing, 1996. Proceedings., International Conference on
Volume 3, 16-19 Sept. 1996 Page(s):359 - 362 vol.3
AbstractPlus | Full Text: PDF(348 KB) IEEE CNF

13. **Public data structures: counters as a special case**
Brit, H.; Moran, S.; Taubenfeld, G.;
Theory of Computing and Systems, 1995. Proceedings., Third Israel Symposium on the
4-6 Jan. 1995 Page(s):98 - 110
AbstractPlus | Full Text: PDF(1096 KB) IEEE CNF

14. **Amortization, lazy evaluation, and persistence: lists with catenation via lazy linking**
Okasaki, C.;
Foundations of Computer Science, 1995. Proceedings., 36th Annual Symposium on
23-25 Oct. 1995 Page(s):646 - 654
AbstractPlus | Full Text: PDF(680 KB) IEEE CNF

15. **A framework to capture dynamic data structures in pointer-based codes**
Corbera, F.; Asenjo, R.; Zapata, E.L.;
Parallel and Distributed Systems, IEEE Transactions on
Volume 15, Issue 2, Feb 2004 Page(s):151 - 166
AbstractPlus | References | Full Text: PDF(947 KB) IEEE JNL

16. **Data structures for associative supercomputers**
Potter, J.L.;

Frontiers of Massively Parallel Computation, 1988. Proceedings., 2nd Symposium on the Frontiers of
10-12 Oct. 1988 Page(s):77 - 84

AbstractPlus | Full Text: PDF(516 KB) IEEE CNF

17. **Structured spreadsheet calculation**
Hassinen, K.; Sajaniemi, J.; Vaisanan, J.;
Languages for Automation: Symbiotic and Intelligent Robots, 1988., IEEE Workshop on
29-31 Aug. 1988 Page(s):129 - 133

AbstractPlus | Full Text: PDF(436 KB) IEEE CNF

18. **On-line maintenance of the four-connected components of a graph**
Kanevsky, A.; Tamassia, R.; Di Battista, G.; Chen, J.;
Foundations of Computer Science, 1991. Proceedings., 32nd Annual Symposium on
1-4 Oct. 1991 Page(s):793 - 801

AbstractPlus | Full Text: PDF(760 KB) IEEE CNF

19. **Pyramid data structures for on-line image progressive transmission**
Rashwan, M.A.; Elsherif, M.S.; Elsayad, A.M.;
Circuits and Systems, 1993., Proceedings of the 36th Midwest Symposium on
16-18 Aug. 1993 Page(s):103 - 106 vol.1

AbstractPlus | Full Text: PDF(272 KB) IEEE CNF

20. **DRUIDS: tools for understanding data structures and algorithms**
Whale, G.;
Multi-Media Engineering Education Proceedings, 1994., IEEE First International Conference on
6-8 July 1994 Page(s):403 - 407

AbstractPlus | Full Text: PDF(984 KB) IEEE CNF

21. **Background memory management for dynamic data structure intensive processing systems**
de Jong, G.; Lin, B.; Verdonck, C.; Wuytack, S.; Catthoor, F.;
Computer-Aided Design, 1995. ICCAD-95. Digest of Technical Papers., 1995 IEEE/ACM International Confer
5-9 Nov. 1995 Page(s):515 - 520

AbstractPlus | Full Text: PDF(596 KB) IEEE CNF

22. **A performance analysis of hierarchical shortest path algorithms**
Fetterer, A.; Shekhar, S.;
Tools with Artificial Intelligence, 1997. Proceedings., Ninth IEEE International Conference on
3-8 Nov. 1997 Page(s):84 - 93

AbstractPlus | Full Text: PDF(884 KB) IEEE CNF

23. **Stack and queue integrity on hostile platforms**
Devanbu, P.T.; Stubblebine, S.G.;
Security and Privacy, 1998. Proceedings. 1998 IEEE Symposium on
3-6 May 1998 Page(s):198 - 206

AbstractPlus | Full Text: PDF(296 KB) IEEE CNF

24. **Progressive shape analysis for real C codes**
Corbera, F.; Asenjo, R.; Zapata, E.;
Parallel Processing, International Conference on, 2001.
3-7 Sept. 2001 Page(s):373 - 380

AbstractPlus | Full Text: PDF(736 KB) IEEE CNF

25. **A tool for data structure visualization and user-defined algorithm animation**
Tao Chen; Sobh, T.;
Frontiers in Education Conference, 2001. 31st Annual
Volume 1, 10-13 Oct. 2001 Page(s):TID - 2-7 vol.1

AbstractPlus | Full Text: PDF(575 KB) IEEE CNF

Indexed by
Inspec

Results for "((data structure<in>metadata)<and>(memory structure<in>metadata))" e-mail

Your search matched 3 of 6032 documents.

A maximum of 100 results are displayed, 25 to a page, sorted by Relevance in Descending order.

» View Session History

» New Search

Modify Search

» Key

((data structure<in>metadata)<and>(memory structure<in>metadata))

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

Check to search only within this results set

Display Format: Citation Citation & Abstract

Select Article Information

1. **Proceedings of the First International Conference on Massively Parallel Computing Systems (MPCS) 1 General-Purpose and Special-Purpose Computing**
Massively Parallel Computing Systems, 1994., Proceedings of the First International Conference on
2-6 May 1994
AbstractPlus | Full Text: PDF(20 KB) IEEE CNF

2. **CRegs: a new kind of memory for referencing arrays and pointers**
Dietz, H.; Chi, C.-H.;
Supercomputing '88. [Vol.1]. Proceedings.
14-18 Nov. 1988 Page(s):360 - 367
AbstractPlus | Full Text: PDF(580 KB) IEEE CNF

3. **VLSI support for a cactus stack oriented memory organization**
Stenstrom, P.;
System Sciences, 1988. Vol.I. Architecture Track, Proceedings of the Twenty-First Annual Hawaii Internationa
Volume 1, 5-8 Jan. 1988 Page(s):211 - 220
AbstractPlus | Full Text: PDF(676 KB) IEEE CNF

Results for **"(build or create data structure and sets of elements<in>metadata)"**

e-mail

Your search matched **0** of **1144315** documents.

A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» View Session History

» New Search

Modify Search

(build 'or' create data structure 'and' sets of elements<in>metadata)

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistance revising your search.

Results for **"(data structure and elements and memory structure<in>metadata)"**

e-mail

Your search matched **0** of **1144303** documents.

A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» View Session History

» New Search

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

Modify Search

(data structure 'and' elements 'and' memory structure<in>metadata)

Check to search only within this results set

Display Format: Citation Citation & Abstract

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistance revising your search.

Results for **"(data structure and query elements<in>metadata)"**

Your search matched **0** of **1144303** documents.

A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» View Session History

» New Search

» **Key**

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

Modify Search

(data structure 'and' query elements<in>metadata)

Check to search only within this results set

Display Format: Citation Citation & Abstract

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistance revising your search.

Results for **"(memory structure and root memory<in>metadata)"**

Your search matched **0** of **1144303** documents.

A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» View Session History

» New Search

Modify Search

(memory structure 'and' root memory<in>metadata)

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistance revising your search.

Results for "(((data structure<in>metadata)<and>(memory structure<in>metadata))<and>(query<in>metadata))" e-mail

Your search matched **0** of **3** documents.

A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» View Session History

» New Search

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

Modify Search

(((data structure<in>metadata)<and>(memory structure<in>metadata))<and>(query<

Check to search only within this results set

Display Format: Citation Citation & Abstract

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistance revising your search.

ip.com IP.com PriorArtDatabase

Secu

April 15, 2005 USPTO

Search

Full Text
Concept
Document ID
Recent Disclosures

Publish

Publish Disclosure

My IP.com

Manage Account
Prior Purchases
Prior Disclosures
Events
Main Page
Support
Logout

Fingerprint Lookup

Lookup

Displaying records #1 through 10 out of 500
(search stopped at 500 hits)

Result # 1 Relevance:

PREVIEW this document

Graphical user interface for editing a palette of col

12-Sep-2000 IPCOM000001487D Er

A graphical user interface is disclosed for interactively editing a colors in response to signals from a user, such as from a pointir interface provides a color space in a color space window on the and draws each color in the palette in ...

Result # 2 Relevance:

PREVIEW this document

Hybrid finite-queue / circular queue data structure

20-Jun-2003 IPCOM000015286D Er

Hybrid finite-queue circular queue data structure This publicatio hybrid data structure that is a cross between a circular queue a growable queue. The data structure starts out as an infinitely gi and when it reaches a certain size, ...

Result # 3 Relevance:

PREVIEW this document

Basic Socket Interface Extensions for IPv6 (RFC25

13-Sep-2000 IPCOM000003139D Er

The de facto standard application program interface (API) for T(applications is the "sockets" interface. Although this API was de Unix in the early 1980s it has also been implemented on a wide Unix systems. TCP/IP applications ...

Result # 4 Relevance:

PREVIEW this document

Cabletron's VLS Protocol Specification (RFC2642)

13-Sep-2000 IPCOM000003230D Er

The Virtual LAN Link State Protocol (VLSP) is part of the InterSv Protocol (ISMP) which provides interswitch communication betw running Cabletron's SecureFast VLAN (SFVLAN) product. VLSP i: determine and maintain a fully connected mesh ...

Result # 5 Relevance:

PREVIEW this document

A Data Restructuring Methodology for Reducing All Cache Lines

20-May-2003 IPCOM000012685D Er

Presented is a methodology for reducing the number of cache li the fields composing a structure, that are accessed during appli execution. This would in turn reduce the number of cache lines, the amount of data, transferred between ...

Result # 6 Relevance:

PREVIEW this document

Basic Socket Interface Extensions for IPv6 (RFC34

18-Mar-2003 IPCOM000011809D Er

The de facto standard Application Program Interface (API) for T applications is the "sockets" interface. Although this API was de Unix in the early 1980s it has also been implemented on a wide

Unix systems. TCP/IP applications ...

Result # 7 Relevance:

PREVIEW

Differential Index Management With Low Priority E Updating in Database Management Systems

15-Feb-2005 IPCOM000057627D Er

A technique is described whereby differential index managemer data files of relational database management systems allows all continually current, without incurring significant updating overh several problems can exist whenever a ...

Result # 8 Relevance:

PREVIEW

Computer Aided Design Data Handler with Logical

30-Mar-2005 IPCOM000115220D Er

This article describes a Computer Aided Design (CAD) data stru system, which manages CAD-related graphic/geometry/other at entities representing certain structures by relational pointers. It functional capability which makes it possible to ...

Result # 9 Relevance:

PREVIEW

Data Structure for Selecting Objects by Name or fr Attributes

19-Mar-2005 IPCOM000104586D Er

Selecting from among a very large set of objects that share ma simplified by allowing the user to select an object either by givi by choosing from among the attributes to arrive at a name. Eac selects an attribute, the remaining ...

Result # 10 Relevance:

PREVIEW

The method of displaying and operating a map bet complicated data structures

23-Apr-2003 IPCOM000012253D Er

This article is to say a way of displaying and operating mapping between complicated data structures. Showing an arrow from a item to a target data item is a general way of displaying the ma hard to show all the arrows at the same ...

Displaying page 1 of 50 < BACK | NEXT >

Search query: data structure

New search | Modify this search | Search within current results

April 15, 2005

USPTO

Search

Full Text
Concept
Document ID
Recent Disclosures

Publish

Publish Disclosure

My IP.com

Manage Account
Prior Purchases
Prior Disclosures
Events
Main Page
Support
Logout

Fingerprint Lookup

Lookup

No records matched your search.

Perhaps you should try a less restrictive query.

Search query: data structure with memory structure

New search | Modify this search

IP.com PriorArtDatabase

April 15, 2005

USPTO

Search

- Full Text
- Concept
- Document ID
- Recent Disclosures

Publish

- Publish Disclosure

My IP.com

- Manage Account
- Prior Purchases
- Prior Disclosures
- Events
- Main Page
- Support
- Logout

Fingerprint Lookup

Lookup

No records matched your search.

Perhaps you should try a less restrictive query.

Search query: data structure with query elements

New search | Modify this search

IP.com PriorArtDatabase

April 15, 2005

USPTO

Search

- Full Text
- Concept
- Document ID
- Recent Disclosures

Publish

- Publish Disclosure

My IP.com

- Manage Account
- Prior Purchases
- Prior Disclosures
- Events
- Main Page
- Support
- Logout

Fingerprint Lookup

Lookup

No records matched your search.

Perhaps you should try a less restrictive query.

Search query: data structure and sets of elements

New search | Modify this search

Google

Web Images Groups News Froogle Local[New!] more »

build or create data structure and sets of elem Search Advanced Search Preferences

"of" is a very common word and was not included in your search. [details]
Lowercase "or" was ignored. Try "OR" to search for either of two terms. [details]
The "AND" operator is unnecessary -- we include all search terms by default. [details]

Web Results **1 - 10** of about **20,800,000** for **build or create data structure and sets of elements**. (**0.32** secon

Press Release Newswire and News Release Distribution - eMediawire
www.emediawire.com/ - Similar pages

CS231 Projects, Spring 2005, Project 1
... a simple implementation of a **data structure** that holds **sets** of integers. ... What happens if someone tries to **create** a **set** with more than 10 **elements** in ...
www.cs.colby.edu/~rjones/courses/cs231/spring2005/projects/project1.html - 16k - Cached - Similar pages

4. Designing an XML **Data Structure**
... when designing an XML **structure** is whether to model a given **data** item as a ... editors tend to make it very easy to find and modify **element data**. ...
java.sun.com/xml/jaxp/dist/ 1.1/docs/tutorial/overview/4_design.html - 17k - Cached - Similar pages

Designing an XML **Data Structure**
... Many XML-aware editors make it easy modify **element data**, ... The section Designing an XML **Data Structure** shows how to **create** an external entity that you ...
java.sun.com/webservices/ docs/1.0/tutorial/doc/IntroXML4.html - 17k - Cached - Similar pages

Data Structures in Java: Part 6, Duplicate **Elements**, Ordered ...
... This is the sixth lesson in a miniseries on Java **data structures** and the ... all constructors must **create** a **set** that contains no duplicate **elements**. ...
www.developer.com/tech/article.php/799661 - 64k - Cached - Similar pages

The global **structure** of an HTML document
... offers advisory information about the **element** for which it is **set**. ... is the idea that block **elements create** "larger" **structures** than inline **elements**. ...
www.w3.org/TR/REC-html40/struct/global.html - 73k - Cached - Similar pages

Carla Corkern - I've Got an SGML Database - Why do I need HyTime?
... for collaborative authoring, and support reuse of semantic **data structures**. ... Most **Element** managers require you to **build** your DTD to not only share ...
www.infoloom.com/IHC96/cc2.htm - 28k - Cached - Similar pages

Contents
... TO2_createKeySet-**Create** a Persistent Key **Set** Collection ... TO2_getPathInfoFor-Retrieve Path Information for a Collection **Structure** ...
publib.boulder.ibm.com/infocenter/tpfhelp/ topic/com.ibm.tpf.doc.4.1-PUT19/gtpc2/gtpc2m02.htm - 84k - Cached - Similar pages

Contents
... IBM Informix Dynamic Server Version 10.0 and CSDK Version 2.90 Documentation **Set** ... **Creating** and Using Extended **Data** Types in Dynamic Server ...
publib.boulder.ibm.com/infocenter/ ids9help/topic/com.ibm.ddi.doc/ddi02.htm - 36k -

Sponsored Links

Data Structure Tutor
Help with Assignments **and** Programs
LinkList, BinaryTree, ADT's, Graphs
www.cstutoring.com

Cached - Similar pages

[PPT] Chapter 1: **Elements** of Database Systems
File Format: Microsoft Powerpoint 97 - View as HTML
... Changes to data structure can be made without influencing functioning of ... Types of Relationships between **Data Sets** (Relationship Cardinality) ...
info.cba.ksu.edu/skovar/844/notes/Chapter1.ppt - Similar pages

Gooooooooogle ▶
Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

Google

Web Images Groups News Froogle Local[New!] more »

data structure and sets of elements and memc Search Advanced Search Preferences

"of" is a very common word and was not included in your search. [details]
The "AND" operator is unnecessary -- we include all search terms by default. [details]

Web Results **1** - **10** of about **6,960,000** for **data structure and sets of elements and memory structure**. (**0.29**

Contents
... |Defining **Data Structure** Parameters in a Prototype or Procedure Interface ... or **Set** Null Indicator); |%OCCUR (**Set**/Get Occurrence of a **Data Structure**) ...
publib.boulder.ibm.com/iseries/ v5r1/ic2924/books/c092508302.htm - 69k - Cached - Similar pages

Contents
... Varying-Length **Data** Type **Structures**. Using a Varying-Length **Structure**. Managing **Memory** for a Varying-Length **Structure** ...
publib.boulder.ibm.com/infocenter/ids9help/topic/com.ibm.dapip.doc/dapip02.htm - 49k - Cached - Similar pages
[More results from publib.boulder.ibm.com]

Sponsored Links

Data Structure
How database-driven apps can overcome telecom's top challenges.
www.mysql.com

Data Structure Tutor
Help with Assignments **and** Programs LinkList, BinaryTree, ADT's, Graphs
www.cstutoring.com

Systems Programming
... the programmer must handle **memory** management for complex **data structures** ... no **memory** available */ p->first = NULL; /* **set** pointer to the first **element** ...
www.csc.calpoly.edu/~fkurfess/Courses/ 317/S05/Slides/3-C-Data-Structures/3-C-Data-Structures.html - 20k - Cached - Similar pages

Bounds on Unnecessary Space Retention by Conservative Garbage ...
... the program relied on deallocating reachable, but unaccessed **memory**. ... Call a **data structure**, ie a **set** of objects, circular if, for any two objects x ...
www.hpl.hp.com/personal/Hans_Boehm/gc/bounds.html - 10k - Cached - Similar pages

Elements of **Data Structures** and Algorithms
... A **structure** is a **set** of relations of some **elements**. Generally, a **data** ... can access only a sequence of **memory** units, and thus any **structure** that is not ...
www.icl.pku.edu.cn/member/ bswen/_old_stuff/art/chapter-4.html - 8k - Cached - Similar pages

Application Development Guide --Core Components
... Remote Procedure Call with Character/Code **Set** Interoperability ... Binding **Data Structure** Size Calculation Macros; Schema Entry **Data Structure** Size ...
www.univie.ac.at/dcedoc/A3U2J/A3U2JM02.HTM - 61k - Cached - Similar pages

Containers In **Memory**: How Big Is Big?
... Each standard container uses a different underlying **memory structure** and ... which still requires at least three pointers per **element** in the **set**.)[3] ...
www.gotw.ca/publications/mill14.htm - 63k - Cached - Similar pages

[PPT] Dynamic **Memory** allocation Linked Lists
File Format: Microsoft Powerpoint 97 - View as HTML
... Dynamic **Memory** allocation Linked Lists. 2. Introduction. dynamic **data structures** - grow and shrink during execution ...
www.ctr.kcl.ac.uk/lectures/Elaheh/note16.ppt - Similar pages

Universal **Data Structure**
... In most cases the LDS will point to the **memory structure** that holds the **data** and ... Section - named **element** that contains a **set** of records and other ...
www.newobjects.com/pages/ndl/ VarDictionary/ref-StructuredConfigurations.htm - 22k - Cached - Similar pages

The **Structure** of a Collection. Ver 1.0+
... of a **element** class to refer to the **structure** of the collection as a whole. ... A collection also has a **set** of **data** fields and member functions that are ...
www.extreme.indiana.edu/ sage/pcxx_ug/subsection3_5_4.html - 10k - Cached - Similar pages

Goooooooooogle ▶
Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

data structure and sets of elements | Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - About Google

Google

Web Images Groups News Froogle Local[New!] more »

data structure and query elements Search Advanced Search Preferences

The "AND" operator is unnecessary -- we include all search terms by default. [details]

Web Results **1 - 10** of about **2,750,000** for **data structure and query elements** . (**0.29** seconds)

JOT: Journal of Object Technology - On Supporting **Structure** ...
... types of the **data** values, and the hierarchical **structure** of the **elements**. ...
The hierarchical **structure** of **elements** in XML certainly called for **query** ...
www.jot.fm/issues/issue_2004_07/column3 - 34k - Cached - Similar pages

VisualAge RPG Language Reference - Contents
... Parameters in a Prototype or Procedure Interface; Defining **Data Structure**
Subfields ... Specifying the Table Element Found in a LOOKUP Operation ...
publib.boulder.ibm.com/infocenter/iadthelp/ topic/com.ibm.etools.iseries.varpg.doc/sc09245102.htm - 72k - Cached - Similar pages

Contents
... |Defining **Data Structure** Parameters in a Prototype or Procedure Interface ...
Specifying the Table Element Found in a LOOKUP Operation ...
publib.boulder.ibm.com/iseries/ v5r1/ic2924/books/c092508302.htm - 69k - Cached - Similar pages

[PDF] Using a Compact Tree to Index and **Query** XML **Data**
File Format: PDF/Adobe Acrobat - View as HTML
... Ctree is an effective **data structure** for managing XML **data**. ... Q, a **query** tree.
Output: A list of **elements** in T that satisfy the Q. ...
www.cobase.cs.ucla.edu/tech-docs/zou/CIKM04.pdf - Similar pages

Wayne's Little **Data Structures** and Algorithms Library
... Many of them are taken verbatim from textbooks on **data structures** and ...
For 16384 **elements** each about 1K in size, it's about 10 times faster than ...
www.cs.toronto.edu/~wayne/libwayne/libwayne.html - 12k - Cached - Similar pages

Query Languages for XML
... For instance, here is an OQL **query** that finds all SECT1 **elements** that ...
The form of the **query** depends on the **data structures** used to implement the ...
mailman.ic.ac.uk/pipermail/ xml-dev/1997-November/001296.html - 12k - Cached - Similar pages

Introduction
... To build up your **query** you simply drag and drop **elements** and attributes among
... the **structure** of the target **data**; that is, the **structure** of the **query** ...
edocs.bea.com/liquiddata/docs81/querybld/intro.html - 30k - Cached - Similar pages

[PPT] XRANK
File Format: Microsoft Powerpoint 97 - View as HTML
... The XML **data** model addresses this limitation by allowing for extensible element
... Exploit XML's tagged and nested **structure** during **query** processing. ...
www.cmpe.boun.edu.tr/courses/ cmpe521/fall2003/aksu-akkus.ppt - Similar pages

dD Range and Segment Tree
... This chapter presents the CGAL range tree and segment tree **data structures**.
... a window **query** is performed, and the **query elements** are given out. ...
www.cgal.org/Manual/doc_html/ cgal_manual/SearchStructures/Chapter_main.html - 37k - Apr 13, 2005 - Cached - Similar pages

Glossary - S
... part of a Structured **Query** Language statement. stack: A **data structure** in which new **elements** are added to and removed from the top of the **structure**. ...
www.absoft.com/Products/Compilers/ C_C++/XLC/docs/glossary/czgs.htm - 24k - Cached - Similar pages

Gooooooooogle ▶
Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

data structure and query elements Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - About Google